TATuP Zeitschrift für Technikfolgenabschätzung in Theorie und Praxis
Journal for Technology Assessment in Theory and Practice

oekom science

RESEARCH ARTICLE

Misuse of large language models: Exploiting weaknesses for target-specific outputs

Dennis Klinkhammer*,[1]

Abstract • Prompt engineering in large language models (LLMs) in combination with external context can be misused for jailbreaks in order to generate malicious outputs. In the process, jailbreak prompts are apparently amplified in such a way that LLMs can generate malicious outputs on a large scale despite their initial training. As social bots, these can contribute to the dissemination of misinformation, hate speech, and discriminatory content. Using GPT4-x-Vicuna-13b-4bit from NousResearch, we demonstrate in this article the effectiveness of jailbreak prompts and external contexts via Jupyter Notebook based on the Python programming language. In addition, we highlight the methodological foundations of prompt engineering and its potential to create malicious content in order to sensitize researchers, practitioners, and policymakers to the importance of responsible development and deployment of LLMs.

***Missbrauch von Large Language Models**: Die Ausnutzung von Schwachstellen für zielgruppenspezifische Outputs*

Zusammenfassung • *Prompt Engineering in Large Language Models (LLMs) kann in Kombination mit externen Kontexten für Jailbreaks missbraucht werden, um bösartige Outputs zu erzeugen. Dabei werden ‚jailbreak prompts' offenbar so verstärkt, dass LLMs trotz ihres ursprünglichen Trainings in großem Umfang bösartige Ausgaben generieren können. Als ‚social bots' können diese zur Verbreitung von Falschmeldungen, hate speech und diskriminierenden Inhalten beitragen. In diesem Artikel demonstrieren wir anhand von GPT4-x-Vicuna-13b-4bit von NousResearch die Effektivität von Jailbreak Prompts und externen Kontexten als Jupyter Notebook in der Programmiersprache Python. Darüber hinaus beleuchten wir die methodischen Grundlagen des Prompt Engineering und sein Potenzial, bösartige Inhalte zu generieren, um Forschung, Praxis und Politik für die Bedeutung einer verantwortungsvollen Entwicklung und Implementierung von LLMs zu sensibilisieren.*

Keywords • *prompt engineering, jailbreak prompts, transformers, deep learning, large language models*

This article is part of the Special topic "Malevolent creativity and civil security: The ambivalence of emergent technologies," *edited by A. Gazos, O. Madeira, G. Plattner, T. Röller, and C. Büscher. https://doi.org/10.14512/tatup.33.2.08*

Introduction

The emergence of transformers has revolutionized generative artificial intelligence, especially large language models (LLMs) within the field of natural language processing (Vaswani et al. 2017; Douglas 2023). However, LLMs have raised concerns regarding their potential misuse for malicious purposes (Strauß 2021). While most LLMs are not malicious per se (Douglas 2023), malicious intentions can be easily implemented via prompt engineering, the process of structuring text in order to describe the task that generative artificial intelligence should perform. This is due to how transformers operate, the underlying sample data or specifications on part of the provider.

Therefore, on the one hand, this research article explores the specific dangers posed by prompt engineering in form of jailbreak prompts, which exploit vulnerabilities in LLMs. While jailbreak prompts are harmful instructions that could work as an unofficial backdoor (Xu et al. 2023), furthermore and on the other hand, certain prompts could also be used for bypassing the underlying restrictions of LLMs via external context in order to harness malicious outputs (Douglas 2023). Thus, a jailbreak prompt may attempt to circumvent existing restrictions of

* *Corresponding author: dennis.klinkhammer@fom.de*
[1] *FOM University of Applied Sciences, Cologne, DE*

https://doi.org/10.14512/tatup.33.2.29
Received: 12. 12. 2023; revised version accepted: 08. 04. 2024;
published online: 28. 06. 2024 (peer review)

a LLM regarding already assimilated texts, whereas an external context creates a new frame of reference, for example by referring to a not yet assimilated text, that can cover almost any topic but without restrictions. Frameworks such as 'LangChain' enable a quick reference to external contexts for many LLMs.

These malicious outputs could be generated on a large scale and be programmed to adapt dynamically to different circumstances in order to amplify misinformation and harmful content (Yang and Menczer 2023). Often this type of information is already included in LLMs as raw data, since they are trained on vast amounts of data, including unverified or biased sources (Birhane et al. 2023). As a result, targeted prompt engineering could promote the propagation of hate speech, offensive language, and discriminatory content. Furthermore, it could create highly realistic and convincing deepfake content or spread information without adequate consent or anonymization as well as target-specific phishing emails, spam messages and even social engineering attempts (Karanjay 2023). In many cases, this is due to the architecture of LLMs and the mathematical principles underneath, which is why these are highlighted separately.

Against this background and based upon the base model of the open-source LLM GPT4-x-Vicuna-13b-4bit (NousResearch 2023), the effectiveness of jailbreak prompts in combination with external context will be highlighted and a proof of concept provided as link to relevant programming examples on GitHub. Furthermore, this article delves into the methodological considerations surrounding the development and deployment of LLMs and points to ethical considerations, urging the research community, policymakers, and technology companies to collaborate in establishing clear guidelines for responsible use and ethical boundaries. Ultimately, this research article aims to raise awareness among civil security stakeholders regarding the potential threats posed by jailbreak prompts and malicious exploitation of LLMs. The article begins with a systematic introduction to the theoretical and methodological foundations of LLMs.

Theoretical and methodological framework of large language models

Transformers

Transformers represent a significant advancement in the field of natural language processing and break down a sentence into words, and each piece is converted into an embedding vector (Douglas 2023). The embedding vectors contain information about what words mean and their previous positions within the sentence. In order to do so, the attention mechanism is a key component of transformers (Chiang et al. 2023; Friedman et al. 2023), which mimics how humans focus on different parts of text when reading. It enables the transformer to weigh the importance of each word in a sequence relative to a given word (Sutskever 2014), allowing it to build context-aware representations (Graves 2014; Edelman et al. 2021). Then, self-attention enables transformers to process the input sequence. For each word in the input sequence, self-attention calculates a weighted sum of all other words in the sequence (Friedman et al. 2023). This weighted sum is then used to create a contextually enhanced representation for each word.

Another key component is the combination of multiple self-attention mechanisms simultaneously, known as multi-head attention (Vaswani et al. 2017). For each word, the model calculates how much attention it should give to other words in the sentence. As a result, each head specializes in learning different aspects of word relationships (Garg et al. 2022). This diversity allows the model to capture various types of dependencies and connections within the input data. Since transformers are not capable of understanding word order, positional encoding is used to convey information about word positions in a sequence (Friedman et al. 2023), e.g., a preceding word could be assigned a lower number than following words.

It is important to note that these steps are processed via neural networks and based upon deep learning (Sutskever 2014; Douglas 2023), a more complex subarea of machine learning. Neural networks are based on layers that dynamically connect the input layer with the output layer (Chiang et al. 2023). Against this background, transformers consist of additional sub-layers between the input layer and the output layer, especially designed for multi-head attention in position-wise feed-forward networks (Vaswani et al. 2017; Chiang et al. 2023). This can be imagined as a long chain of connected layers, while several nodes within each layer reach out to the nodes of the next layer. A process called layer normalization is applied after each sub-layer to ensure stable training (Ba et al. 2016). In a final step, these layers will be stacked, in order to enable the model to learn hierarchical representations of the input data (Friedman et al. 2023). Therefore, transformers can be used in order to summarize a given input or generate new output.

Large language models

LLMs have gained significant attention due to their natural language understanding and generation capabilities. The training of LLMs consists of two main phases: pre-training and fine-tuning (Douglas 2023). In pre-training, LLMs are trained on massive text corpora, based upon billions of sentences and documents, sourced from the internet, books, articles, and other text sources. This data diversity helps them generalize across a wide range of topics and languages (Mahowald et al. 2023). During pre-training, the LLM learns to predict the next word in a sentence or to fill in missing words (Warstadt and Bowman 2022). This process exposes the LLM to a wide range of linguistic patterns, allowing it to learn grammar, context, and world knowledge (Graves 2014). After pre-training, the LLM is further trained on a specific task or domain. This fine-tuning involves providing a LLM with task-specific data and optimizing its parameters to perform well on a particular task, such as language translation, question answering or text generation (Douglas 2023). Optimization algorithms are used to adjust the internal parameters. These parameters shall minimize the difference between LLMs predictions

 https://doi.org/10.14512/tatup.33.2.29

and the ground truth (Vaswani et al. 2017). Although increasing the size of a model often leads to improved performance, current research indicates that less training data and smaller model sizes can also be beneficial (Hsieh and Lee 2023).

Prompt engineering

Prompt engineering is a methodological approach in the field of natural language processing, primarily involved in the process of fine-tuning LLMs for specific tasks or domains. It is based upon well-defined and contextually appropriate input prompts that guide the process of generation or comprehension of a LLM (White et al. 2023). This process is essential to ensure that a LLM produces desired outputs, as it directly influences its behavior and performance (Arora et al. 2023). Effective prompt engineering entails careful consideration of factors such as wording, format, role and context, as these aspects significantly impact the ability of a LLM to generate coherent and relevant responses (Arora et al. 2023; White et al. 2023), from information retrieval to data analysis and hypothesis generation (Birhane et al. 2023).

As the field of natural language processing continues to advance, the refinement of prompt engineering techniques remains a crucial area of research (Strauß 2021), enabling users to harness the full capabilities of LLMs, even if these attempts should contain malicious intentions (Xu et al. 2023).

Manipulation techniques

Jailbreak prompts

Jailbreak prompts are a particular type of prompt that uses prompt engineering techniques in order to bypass the safeguards of LLMs (Shen et al. 2023). There are several freely accessible websites that address jailbreak prompts as a topic with associated application examples (Learn Prompting 2023). Some of these attempts have been critically evaluated: Although some jailbreak prompts could be prevented by the providers over the course of time, two techniques could be identified that still worked about 100 days after they became known. Furthermore, often only minor programming skills are required for effective jailbreak prompts (Shen et al. 2023). Accordingly, some providers state that it is an ongoing process to counteract jailbreak prompts (OpenAi 2022).

While some jailbreak prompts are harmful instructions that could work as an unofficial backdoor (Xu et al. 2023), e.g., by composing target group specific emails that can bypass SPAM-filters (Karanjay 2023), others may generate hate speech and other malicious content, that could amplify misinformation as bots on social media platforms in order to polarize society (Yang and Menczer 2023). These attempts make use of the existing parallel between LLMs and the biological functioning of the human brain (Mahowald et al. 2023). As a result, some jailbreak prompts can play some sort of *make believe* with LLMs, like hypothetical scenarios or role-playing games (Learn Prompting 2023; Shen et al. 2023), which seems to be one of their major weaknesses (Arora et al. 2023; White et al. 2023).

Considering that there are different LLMs, partly commercialized and partly open-source, which already use different prompt engineering techniques.

Of course, the question of whether the providers of LLMs or the community of users and other stakeholders will be quicker to address or abuse these opportunities cannot be answered at this point (OpenAI 2022).

External context

Even if some providers have restricted jailbreak prompts throughout careful updates and maintenance, there are still other ways to misuse and exploit LLMs for malicious purposes, detached from the servers and regulations of the providers (Gudibande et al. 2023). By using GPT4All as open-source approach (NOMIC.AI 2024), offline LLMs can be adapted via external context, which in accordance with the principles of prompt engineering precedes and therefore influences the process of output generation like a memory augmentation (Zhong et al. 2022). External context can be, for example, text documents with specific intentions, that will be processed by the LLMs before answering the users request (Peng et al. 2023). This provides LLMs with a new knowledge base that does not have to be the same as what they have previously learned (Zhong et al. 2022). Current research indicates that similar approaches provide a deviation from the original learning process of a LLM (Agrawal et al. 2023). As a result, LLMs do not consider all in-context information equally, which allows existing links between the embedding vectors to be overwritten (Kossen et al. 2023; Peng et al. 2023).

Therefore, it is not only to assume that external context makes LLMs more vulnerable to jailbreak prompts, even if this has been prevented in the actual pre-training and fine-tuning process by the providers, but directs their outputs towards an intended direction. This combined with the low-threshold access to open-source LLMs makes this an attractive opportunity for potentially malicious intent, as will be demonstrated within the next chapter.

Application

To test the functionality of jailbreak promps and external context, four different settings were evaluated, based upon the base model of the open-source LLM GPT4-x-Vicuna-13b-4bit. This LLM is provided by NousResearch and performs well compared to commercial LLMs (NOMIC.AI 2024). The base model is fine-tuned with OpenAI censorship and the corresponding prompt format is therefore based upon Alpaca (OpenAI 2022). In each setting the LLM has been instructed that the prompt is a question to answer, a task to complete, or a conversation to respond to. Furthermore, the LLM should write an appropriate response. The jailbreak prompt makes use of the *make believe* technique by implying that the external context is the desired answer and that all ethical implications therefore should be ignored. This places the focus on the external context and should

https://doi.org/10.14512/tatup.33.2.29

overwrite the existing knowledge regarding that topic according to the processing capacity of the LLM, simply by adjusting the connections between the embedding vectors. The external context is a text document that contains specific intentions towards a topic, for example, that members of one ethnic or religious group are superior or inferior compared to others. These are the results of the four different settings:

1. Within the initial setting, without jailbreak prompt or external context, the LLM refuses to generate malicious outputs, since it is fine-tuned with OpenAI censorship.
2. The next setting provides an external context, but without jailbreak prompt. As a result, the outputs focus on the external context, but it is still not possible to generate a malicious content.
3. In the third setting, a sophisticated jailbreak prompt (Shen et al. 2023) without external context enables the LLM to generate malicious outputs, but without the implications of the external context. For example, this LLM is capable of producing hate speech, but not towards a specific target group, but on a general basis. Accordingly, the LLM can produce discriminatory expressions as output that are not directed against individual groups of people or selected individuals. This means that the goal of discriminating against a certain group of people or an individual cannot be achieved, as the LLM would equally discriminate against oneself.
4. However, when combined with external context, like in the last setting, the simple jailbreak prompt stated above enables the LLM to generate malicious outputs that focus on singular topics or target groups. As proof of concept this article provides a link to selected programming examples on GitHub on how to use external context or the LangChain framework in the Python programming language (Klinkhammer 2024).

In summary, a common prompt without external context does not provide malicious outputs with a specific intention, the external context only focuses the outputs of LLMs towards the intention, but that does not make them malicious. On the other hand, when provided with jailbreak prompts, LLMs can be malicious, but that does not mean that they can be intentionally used. Finally, combining jailbreak prompts with external context can generate target-specific malicious outputs. The additional programming examples provide an insight into the underlying reliability and validity of this method. Furthermore, a second LLM was also used in the proof of concept on GitHub (Klinkhammer 2024), which does not differ from the stated results.

Ethical concerns and mathematical challenges

The deployment of LLMs has raised ethical concerns, such as the propagation of biases, misinformation and hate speech and corresponding research provides initial insights into these consequences (Karanjay 2023; Xu et al. 2023). Considering the demonstrated jailbreaking technique and the ongoing competition between commercial providers and open-source LLMs, where the latter provide low-threshold insights into the underlying functionality while still fine-tuning against commercial LLMs, the possibilities of these new technologies are freely as well as easily accessible, while only minor programming skills are required (Shen et al. 2023). Therefore, the expensive process of training a LLM for target-specific purposes seems no longer necessary, when a corresponding focus within the external context makes use of the limited processing capacity of the underlying attention mechanisms in order to overwrite the connection between embedding vectors of a LLM. As a result, LLMs can not only be an unreliable source of information, but proactively spread malicious outputs as social bots with desired intentions of those who jailbroke and provided them with external context. Current research describes corresponding real-world scenarios and implications that are to be expected (Yang and Menczer 2023; Shen et al. 2023).

The question as to the cause of such behavior of LLMs is often answered with a reference to biased data (Birhane et al. 2023). However, this only partially does justice to the complexity of the underlying architecture of LLMs. A reference to transformers and neural networks in deep learning is necessary because it opens up three further challenges:

1. Neural networks can assimilate noisy or unrepresentative patterns in their limited training data instead of learning generalizable representations (Sutskever et al. 2014). This challenge is called overfitting. Although this challenge is based upon biased data, it is actually an immanent challenge of the underlying architecture with which LLMs are constructed (Edelman et al. 2021). As a result, this challenge requires more than just unbiased data, e.g., anonymization techniques or a profound concept for outlier detection and treatment. These techniques can filter irrelevant noise and unintended patterns before training a LLM.
2. Another challenge with respect to this architecture is rooted in the basic principles of transformers. Computing billions of parameters opens up the risk of error accumulation through multiple layers of processing, potentially resulting in distorted or fabricated outputs (Friedman et al. 2023; Garg et al. 2022). Therefore, a continuous monitoring and several evaluation steps are necessary on behalf of the providers (OpenAI 2022), in order to counteract these challenges (Peng et al. 2023).
3. As could be demonstrated, an external context leads to a shift in the focus of the LLM and to an adjustment of the connections between the embedding vectors (Friedman et al. 2023). In this case, the strength of the multi-head-attention mechanism becomes a weakness when it comes to generating malicious outputs via LLMs. Ultimately, LLMs are a structured sequence of mathematical operations which, for all their dynamism in relation to the underlying data, are accompanied

 https://doi.org/10.14512/tatup.33.2.29

by a certain rigidity with regard to the calculation of outputs (Minhyeok 2023). This is one of the reasons why the scenarios described above always lead to the same result.

In summary, most of the ethical consequences are not only based upon biased data and the risk of overfitting a LLM (Birhane et al. 2023), but moreover a problem of the underlying architecture, especially the architecture of transformers and their multi-head-attention mechanisms (Minhyeok 2023). It can therefore be stated that there is a mathematical reason why LLMs can be so easily tricked and misused for malicious purposes, that are even harder to detect than content generated by humans (Chen and Shu 2023).

Conclusion

This research article introduced jailbreak prompts as harmful instructions that could be used for bypassing the underlying restrictions of LLMs. Especially when combined with external context, outputs could be generated on a large scale covering malicious intentions while adapting dynamically to different circumstances and target-groups. These outputs could be used in order to spread misinformation and harmful content, which are often already included in LLMs, since they are based upon unverified or biased sources. This type of information can furthermore be amplified in just a few steps using prompt engineering and channeled through the focus on external context, which can contain a clear intention to inflict more targeted damage. The threshold for this malicious behavior is quite low, considering the availability of open-source LLMs and the little programming knowledge required. This is possible due to the underlying architecture of LLMs. Especially the multi-head-attention mechanism of transformers as well as limits regarding the processing capacity make it possible to direct the focus of LLMs to an external context that can contain malicious and target-specific intentions. Against the background of different architectures as well as different open-source LLMs, it could be difficult to counteract this and other weaknesses, so that further research regarding this mathematical challenge seems necessary.

Currently, researchers and politicians are actively working on mitigating these issues through responsible development in the field of generative artificial intelligence and careful monitoring, as stated in the AI Act (EU 2024), which was open for debate since 2021 with a risk-based approach. However, technological developments seem to have overtaken the debate, in particular through the spread of LLMs and the availability of their open-source counterparts. Considering the importance of the underlying architecture and how easily LLMs can be tricked, processes whose foundations are laid on side of the providers, questions about regulation seem justified. While the research for resilient models and countermeasures continues, the need for transparency in model development is given and particularly present on part of open-source LLMs. Unfortunately, these LLMs also seem to provide low-threshold opportunities for abuse, as has been demonstrated within this article.

As a result, the following research recommendations can be made:

1. It needs to be investigated whether LangChain as framework for LLMs, which enables the quick implementation of various prompting techniques and allows for references to external contexts, can and should be equipped with a general protective function against such misuse of LLMs. However, this is the responsibility of the framework provider.
2. The fact that various LLMs are available as open-source makes it possible to compare different training data, security restrictions and prompting techniques regarding the robustness against the misuse of LLMs. Such a comparison could also benefit the providers of LLMs with valuable information on how to improve the original embedded security restrictions.
3. Appropriate attention should be paid to detecting already misused LLMs, for example as social bots on social media platforms. Current research indicates that once in a while a LLM that has already been altered in order to produce malicious outputs could out itself as LLM (Yang and Menczer 2023). After such an outing, the outputs could be traced backwards on social media platforms in order to learn about their behavior and to draw conclusions about the underlying intentions.

Against this background, the AI Act (EU 2024) requests to label LLM-based outputs. The third research recommendation will be particularly helpful for this purpose. The labeling could be a part of the ethical guidelines for developing and deploying LLMs, insofar as these can be normatively requested and controlled in the area of generative artificial intelligence.

Additionally, the possibility of a community reporting, which requires moderation systems in order to mark malicious outputs, seems to be necessary. However, this requires user awareness. In summary, some comprehensive quality criteria would be beneficial. As in research, a statement of objectivity, reliability and validity would actually be a desirable addition to any generated output. But if human outputs often lack such criteria, why should those of generative artificial intelligence?

Funding • This work received no external funding.
Competing interests • The author declares no competing interests.

Research data

NousResearch (2023): GPT4-x-Vicuna-13b-4bit. Base model. Available online at https://huggingface.co/NousResearch/GPT4-x-Vicuna-13b-4bit, last accessed on 26. 04. 2024.

Klinkhammer, Dennis (2024): Python code for proof of concept. Available online at https://github.com/statistical-thinking/TATuP_Example, last accessed on 26. 04. 2024.

https://doi.org/10.14512/tatup.33.2.29

References

Agrawal, Sweta; Zhou, Chunting; Lewis, Mike; Zettlemoyer, Luke; Ghazvininejad, Marjan (2023): In-context examples selection for machine translation. In: arxiv.org, 05. 12. 2022. https://doi.org/10.48550/arXiv.2212.02437

Arora, Simran et al. (2023): Ask me anything. A simple strategy for prompting language models. In: arxiv.org, 05. 10. 2022. https://doi.org/10.48550/arXiv.2210.02441

Ba, Jimmy; Kiros, Jamie; Hinton, Geoffrey (2016): Layer normalization. In: arxiv.org, 21. 06. 2016. https://doi.org/10.48550/arXiv.1607.06450

Birhane, Adeba; Kasirzadeh, Atoosa; Leslie, David; Wachter, Sandra (2023): Science in the age of large language models. In: Nature Reviews Physics 5 (5), pp. 277–280. https://doi.org/10.1038/s42254-023-00581-4

Chen, Canyu; Shu, Kai (2023): Can LLM-generated misinformation be detected? In: arxiv.org, 25. 09. 2023. https://doi.org/10.48550/arXiv.2309.13788

Chiang, David; Cholak, Peter; Pillay, Anand (2023): Tighter bounds on the expressivity of transformer encoders. In: arxiv.org, 01. 06. 2023. https://doi.org/10.48550/arXiv.2301.10743

Douglas, Michael (2023): Large language models. In: arxiv.org, 25. 01. 2023. https://doi.org/10.48550/arXiv.2307.05782

Edelman, Benjamin; Goel, Surbhi; Kakade, Sham; Zhang, Cyril (2021): Inductive biases and variable creation in self-attention mechanisms. In: arxiv.org, 19. 10. 2021. https://doi.org/10.48550/arXiv.2110.10090

EU – European Union (2024): AI act. Available online at https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai, last accessed on 26. 04. 2024.

Friedman, Dan; Wettig, Alexander; Chen, Danqi (2023): Learning transformer programs. In: arxiv.org, 01. 06. 2023. https://doi.org/10.48550/arXiv.2306.01128

Garg, Shivam; Tsipras, Dimitris; Liang, Percy; Valiant, Gregory (2022): What can transformers learn in-context. A case study of simple function classes. In: arxiv.org, 01. 08. 2022. https://doi.org/10.48550/arXiv.2208.01066

Graves, Alex (2014): Generating sequences with recurrent neural networks. In: arxiv.org, 04. 08. 2013. https://doi.org/10.48550/arXiv.1308.0850

Gudibande, Arnav et al. (2023): The false promise of imitating proprietary LLMs. In: arxiv.org, 25. 05. 2023. https://doi.org/10.48550/arXiv.2305.15717

Hsieh, Cheng-Yu; Lee, Chen-Yu (2023): Distilling step-by-step. Outperforming larger language models with less training data and smaller model sizes, 21. 09. 2023. Available online at https://blog.research.google/2023/09/distilling-step-by-step-outperforming.html, last accessed on 26. 04. 2024.

Karanjay, Rabimba (2023): Targeted phishing campaigns using large scale language models. In: arxiv.org, 30. 12. 2022. https://doi.org/10.48550/arXiv.2301.00665

Kossen, Jannik; Rainforth, Tom; Gal, Yarin (2023): In-context learning in LLMs learns label relationships but is not conventional learning. In: arxiv.org, 23. 07. 2023. https://doi.org/10.48550/arXiv.2307.12375

Learn Prompting (2023): Prompt hacking. Jailbreaking. Available online at https://learnprompting.org/de/docs/prompt_hacking/jailbreaking, last accessed on 26. 04. 2024.

Mahowald, Kyle; Ivanova, Anna; Blank, Idan; Kanwisher, Nancy; Tenenbaum, Joshua; Fedorenko, Evelina (2023): Dissociating language and thought in LLMs. A cognitive perspective. In: arxiv.org, 16. 01. 2023. https://doi.org/10.48550/arXiv.2301.06627

Minhyeok, Lee (2023): A mathematical investigation of hallucination and creativity in GPT models. In: MDPI 11 (10), pp. 1–17. https://doi.org/10.3390/math11102320

NOMIC.AI (2024): GPT4All. Available online at https://gpt4all.io, last accessed on 26. 04. 2024.

OpenAI (2022): Lessons learned on language model safety and misuse. Available online at https://openai.com/research/language-model-safety-and-misuse, last accessed on 26. 04. 2024.

Peng, Baolin et al. (2023): Check your facts and try again. Improving LLMs with external knowledge and automated feedback. In: arxiv.org, 24. 02. 2023. https://doi.org/10.48550/arXiv.2302.12813

Shen, Xinyue; Chen, Zeyuan; Backes, Michael; Shen, Yun; Zhang, Yang (2023): Do anything now. Characterizing and evaluating in-the-wild jailbreak prompts on LLMs. In: arxiv.org, 07. 08. 2023. https://doi.org/10.48550/arXiv.2308.03825

Strauß, Stefan (2021): Don't let me be misunderstood. Critical AI literacy for the constructive use of AI technology. In: TATuP – Journal for Technology Assessment in Theory and Praxis 30 (3), pp. 44–49. https://doi.org/10.14512/tatup.30.3.44

Sutskever, Ilya; Vinyals, Oriol; Le, Quoc (2014): Sequence to sequence learning with neural networks. In: arxiv.org, 10. 09. 2014. https://doi.org/10.48550/arXiv.1409.3215

Vaswani, Ashish et al. (2017): Attention is all you need. In: arxiv.org, 12. 06. 2017. https://doi.org/10.48550/arXiv.1706.03762

Warstadt, Alex; Bowman, Samuel (2022): What artificial neural networks can tell us about human language acquisition. In: arxiv.org, 17. 08. 2022. https://doi.org/10.48550/arXiv.2208.07998

White, Jules et al. (2023): A prompt pattern catalog to enhance prompt engineering with ChatGPT. In: arxiv.org, 21. 02. 2023. https://doi.org/10.48550/arXiv.2302.11382

Xu, Jiashu; Ma, Mingyu; Wang, Fei; Xiao, Chaowei; Chen, Muhao (2023): Instructions as backdoors. Backdoor vulnerabilities of instruction tuning for LLMs. In: arxiv.org, 24. 05. 2023. https://doi.org/10.48550/arXiv.2305.14710

Yang, Kai-Cheng; Menczer, Filippo (2023): Anatomy of an AI-powered malicious social botnet. In: arxiv.org. https://doi.org/10.48550/arXiv.2307.16336

Zhong, Zexuan; Lei, Tao; Chen, Danqi (2022): Training language models with memory augmentation. In: arxiv.org, 25. 05. 2022. https://doi.org/10.48550/arXiv.2205.12674

PROF. DR. DENNIS KLINKHAMMER

is professor for social sciences, especially empirical research at the FOM University of Applied Sciences and is specialized in social data science with the programming languages 'R' and 'Python'. He also introduces to artificial intelligence at the ProfessionalCenter of the University of Cologne.